





```
1 0001 0 MODULE LIB$DELETE_FILE (,XTITLE 'Delete one or more files'
2 0002 0 IDENT = '1-006' ! File: LIBDELETE.B32 Edit: BLS0331
3 0003 0 ) =
4 0004 1 BEGIN
5 0005 1
6 0006 1 *****
7 0007 1 *
8 0008 1 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
9 0009 1 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
10 0010 1 * ALL RIGHTS RESERVED.
11 0011 1 *
12 0012 1 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
13 0013 1 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
14 0014 1 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
15 0015 1 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
16 0016 1 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
17 0017 1 * TRANSFERRED.
18 0018 1 *
19 0019 1 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
20 0020 1 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
21 0021 1 * CORPORATION.
22 0022 1 *
23 0023 1 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
24 0024 1 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
25 0025 1 *
26 0026 1 *
27 0027 1 *****
28 0028 1
29 0029 1
30 0030 1 ++
31 0031 1 FACILITY: General Utility Library
32 0032 1
33 0033 1 ABSTRACT:
34 0034 1
35 0035 1 LIB$DELETE_FILE deletes one or more files.
36 0036 1
37 0037 1 ENVIRONMENT: User mode - AST reentrant
38 0038 1
39 0039 1 AUTHOR: Steven B. Lionel, CREATION DATE: 13-July-1982
40 0040 1
41 0041 1 MODIFIED BY:
42 0042 1
43 0043 1 1-001 - Original. SBL 13-July-1982
44 0044 1 1-002 - Add related-file-spec argument, and pass FAB to confirm-routine.
45 0045 1 SBL 1-Oct-1982
46 0046 1 1-003 - Pass error source code to error-routine. SBL 19-Nov-1982
47 0047 1 1-004 - Update for new argument to LIB$FILE_SCAN. BLS 6-FEB-1984
48 0048 1 1-005 - Add new argument for filescan context. BLS 5-MAR-1984
49 0049 1 1-006 - Parse null string to deallocate rms context after calling
50 0050 1 file_scan. BLS 9-JUL-1984
51 0051 1 !--
52 0052 1
```



LIB\$DELETE\_FILE Delete one or more files  
1-006 Declarations

L 14  
16-Sep-1984 00:48:36  
14-Sep-1984 12:38:38

VAX-11 Bliss-32 V4.0-742  
[LIBRTL.SRC]LIBDELETE.B32;1

Page 2  
(2)

```

54 0053 1 %SBTTL 'Declarations'
55 0054 1
56 0055 1 PROLOGUE FILE:
57 0056 1
58 0057 1
59 0058 1 REQUIRE 'RTLIN:LIBPROLOG';
60 0129 1 ! Switches, PSECTS, macros, etc.
61 0130 1
62 0131 1 LINKAGES:
63 0132 1
64 0133 1 NONE
65 0134 1
66 0135 1 TABLE OF CONTENTS:
67 0136 1
68 0137 1
69 0138 1 FORWARD ROUTINE
70 0139 1 LIB$DELETE_FILE,
71 0140 1 DO DELETE: NOVALUE,
72 0141 1 DELETE_ERROR: NOVALUE,
73 0142 1 DELETE_HANDLER;
74 0143 1 ! Delete one or more files
75 0144 1 ! Delete a file
76 0145 1 ! Error routine
77 0146 1 ! Condition handler
78 0147 1
79 0148 1
80 0149 1 MACROS:
81 0150 1
82 0151 1 NONE
83 0152 1
84 0153 1 EQUATED SYMBOLS:
85 0154 1
86 0155 1 NONE
87 0156 1
88 0157 1 FIELDS:
89 0158 1
90 0159 1 NONE
91 0160 1
92 0161 1 OWN STORAGE:
93 0162 1
94 0163 1
95 0164 1 EXTERNALS:
96 0165 1
97 0166 1 EXTERNAL ROUTINE
98 0167 1 LIB$ANALYZE_SDESC_R2: LIB$ANALYZE_SDESC_R2$LINKAGE,
99 0168 1 LIB$FILE_SCAN,
100 0169 1 LIB$COPY_DXDX,
101 0170 1 LIB$COPY_R_DX,
102 0171 1 LIB$SIG_TO_RET;
103 0172 1 EXTERNAL LITERAL
104 0173 1 LIB$ERRROUCAL,
LIB$INVFILSPE;
```

```
106 0174 1 %SBTTL 'LIB$DELETE_FILE - Delete one or more files'
107 0175 1 GLOBAL ROUTINE LIB$DELETE_FILE (
108 0176 1     FILESPEC: REF BLOCK [, BYTE],           | File to delete
109 0177 1     DEFAULT_FILESPEC: REF BLOCK [, BYTE], | Default file specification
110 0178 1     RELATED_FILESPEC: REF BLOCK [, BYTE], | Related file specification
111 0179 1     SUCCESS_ROUTINE,                     | Called on successful delete
112 0180 1     ERROR_ROUTINE,                       | Called on error
113 0181 1     CONFIRM_ROUTINE,                    | Called for confirmation
114 0182 1     USER_ARG,                          | User argument
115 0183 1     RESULTANT_NAME: REF BLOCK [, BYTE], | Returned filespec
116 0184 1     FILE_SCAN_CONTEXT                   | Lib$file_scan context
117 0185 1 ) =
118 0186 1
119 0187 1 ++
120 0188 1 FUNCTIONAL DESCRIPTION:
121 0189 1
122 0190 1     LIB$DELETE_FILE deletes one or more files. It is similar in
123 0191 1     function to the DCL DELETE command. The specification
124 0192 1     of the file(s) to be deleted may include wild cards.
125 0193 1
126 0194 1
127 0195 1 CALLING SEQUENCE:
128 0196 1
129 0197 1     ret_status.wlc.v = LIB$DELETE_FILE (
130 0198 1         FILESPEC.rt.dx,
131 0199 1         [, [DEFAULT_FILESPEC.rt.dx]
132 0200 1         [, [RELATED_FILESPEC.rt.dx]
133 0201 1         [, [SUCCESS_ROUTINE.szem.r]
134 0202 1         [, [ERROR_ROUTINE.fzemlc.r]
135 0203 1         [, [CONFIRM_ROUTINE.fzemlc.r]
136 0204 1         [, [USER_ARG.rz]
137 0205 1         [, [RESULTANT_NAME.wt.dx]
138 0206 1         [, [FILE_SCAN_CONTEXT] ]]]]]])
139 0207 1
140 0208 1 FORMAL PARAMETERS:
141 0209 1
142 0210 1     FILESPEC - The file specification of the file(s)
143 0211 1             to be deleted. Passed by descriptor.
144 0212 1             The specification may include wild cards.
145 0213 1
146 0214 1     DEFAULT_FILESPEC - The default file specification of the
147 0215 1             file(s) to be deleted. Passed by descriptor.
148 0216 1             If omitted, the default is the null string.
149 0217 1
150 0218 1     RELATED_FILESPEC - The related file specification of the
151 0219 1             files to be deleted. Passed by descriptor.
152 0220 1             If omitted, the default is the null string.
153 0221 1             "Input file parsing" is used.
154 0222 1
155 0223 1     SUCCESS_ROUTINE - The entry mask of a routine to call for
156 0224 1             each successful delete, passed by reference.
157 0225 1             The calling format of the SUCCESS_ROUTINE is
158 0226 1             as follows:
159 0227 1             CALL SUCCESS_ROUTINE (
160 0228 1                 filespec.rt.ds,
161 0229 1                 user_arg.rz)
162 0230 1
```

LIB\$DELETE\_FILE Delete one or more files  
1-006 LIB\$DELETE\_FILE - Delete one or more files

N 14  
16-Sep-1984 00:48:36 VAX-11 Bliss-32 V4.0-742  
14-Sep-1984 12:38:38 [LIBRTL.SRC]LIBDELETE.B32:1

Page 4  
(3)

163 0231 1  
164 0232 1  
165 0233 1  
166 0234 1  
167 0235 1  
168 0236 1  
169 0237 1  
170 0238 1  
171 0239 1  
172 0240 1  
173 0241 1  
174 0242 1  
175 0243 1  
176 0244 1  
177 0245 1  
178 0246 1  
179 0247 1  
180 0248 1  
181 0249 1  
182 0250 1  
183 0251 1  
184 0252 1  
185 0253 1  
186 0254 1  
187 0255 1  
188 0256 1  
189 0257 1  
190 0258 1  
191 0259 1  
192 0260 1  
193 0261 1  
194 0262 1  
195 0263 1  
196 0264 1  
197 0265 1  
198 0266 1  
199 0267 1  
200 0268 1  
201 0269 1  
202 0270 1  
203 0271 1  
204 0272 1  
205 0273 1  
206 0274 1  
207 0275 1  
208 0276 1  
209 0277 1  
210 0278 1  
211 0279 1  
212 0280 1  
213 0281 1  
214 0282 1  
215 0283 1  
216 0284 1  
217 0285 1  
218 0286 1  
219 0287 1

# ERROR\_ROUTINE

filespec - The RMS resultant file specification of the file being deleted. If RESULTANT\_NAME was specified, it is used to pass the string to SUCCESS\_ROUTINE. Otherwise, a class S, type T string is passed.

user\_arg - The value of user\_arg passed to LIB\$DELETE\_FILE is passed to SUCCESS\_ROUTINE using the same mechanism as was used to pass it to LIB\$DELETE\_FILE

- The entry mask of a routine to call when an error is detected, passed by reference. The function value returned by the routine determines whether or not more files will be processed.

The calling format of the ERROR\_ROUTINE is as follows:

```
ret_status.wlc.v = ERROR_ROUTINE (  
    filespec.rt.ds,  
    RMS_sts.rlc.r  
    RMS_stv.rlc.r,  
    error_source.rl.r,  
    user_arg.rz)
```

filespec - The RMS resultant file specification of the file being deleted. If RESULTANT\_NAME was specified, it is used to pass the string to ERROR\_ROUTINE. Otherwise, a class S, type T string is passed.

RMS\_sts - The primary condition code which describes the error that occurred.

RMS\_stv - The secondary condition code which describes the error that occurred.

error\_source - An integer code that indicates at what point the error was found. The values are:  
0 = Error searching for filespec  
1 = Error deleting file

user\_arg - The value of user\_arg passed to LIB\$DELETE\_FILE is passed to ERROR\_ROUTINE using the same mechanism as was used to pass it to LIB\$DELETE\_FILE



```

: 220 0288 1
: 221 0289 1
: 222 0290 1
: 223 0291 1
: 224 0292 1
: 225 0293 1
: 226 0294 1
: 227 0295 1
: 228 0296 1
: 229 0297 1
: 230 0298 1
: 231 0299 1
: 232 0300 1
: 233 0301 1
: 234 0302 1
: 235 0303 1
: 236 0304 1
: 237 0305 1
: 238 0306 1
: 239 0307 1
: 240 0308 1
: 241 0309 1
: 242 0310 1
: 243 0311 1
: 244 0312 1
: 245 0313 1
: 246 0314 1
: 247 0315 1
: 248 0316 1
: 249 0317 1
: 250 0318 1
: 251 0319 1
: 252 0320 1
: 253 0321 1
: 254 0322 1
: 255 0323 1
: 256 0324 1
: 257 0325 1
: 258 0326 1
: 259 0327 1
: 260 0328 1
: 261 0329 1
: 262 0330 1
: 263 0331 1
: 264 0332 1
: 265 0333 1
: 266 0334 1
: 267 0335 1
: 268 0336 1
: 269 0337 1
: 270 0338 1
: 271 0339 1
: 272 0340 1
: 273 0341 1
: 274 0342 1
: 275 0343 1
: 276 0344 1

```

# CONFIRM\_ROUTINE

If ERROR\_ROUTINE returns a success status, then processing of files will continue. If a failure status is returned, then processing will cease immediately and LIB\$DELETE\_FILE will return with the error status.

If ERROR\_ROUTINE is not specified, LIB\$DELETE\_FILE will return to its caller with the most severe of the error statuses encountered while deleting the files. Otherwise, if ERROR\_ROUTINE is called for an error, the success status LIB\$ERRROUCAL is returned. Note that ERROR\_ROUTINE is not called for errors related to string copying.

- The entry mask of a routine to call before each file is deleted, passed by reference. The function value returned by the routine determines whether or not the file will be deleted.

The calling format of CONFIRM\_ROUTINE is as follows:

```

ret_status.wlc.v = CONFIRM_ROUTINE (
    filespec.rt.ds,
    FAB.rr.r,
    user_arg.rz)

```

filespec - The RMS resultant file specification of the file being deleted. If RESULTANT\_NAME was specified, it is used to pass the string to CONFIRM\_ROUTINE. Otherwise, a class S, type T string is passed.

FAB - The address of the RMS FAB that describes the file being deleted. You may perform an RMS \$OPEN on the FAB to obtain file attributes you need to determine whether the file should be deleted, but you must close the file with \$CLOSE before returning to LIB\$DELETE\_FILE.

user\_arg - The value of user\_arg passed to LIB\$DELETE\_FILE is passed to CONFIRM\_ROUTINE using the same mechanism as was used to pass it to LIB\$DELETE\_FILE

If CONFIRM\_ROUTINE returns success, the file is then deleted, otherwise that file is not deleted.

LIB\$DELETE\_FILE Delete one or more files  
1-006

LIB\$DELETE\_FILE - Delete one or more files

C 15

16-Sep-1984 00:48:36

14-Sep-1984 12:38:38

VAX-11 Bliss-32 V4.0-742

[LIBRTL.SRC]LIBDELETE.B32;1

Page 6  
(3)

```
277 0345 1 1 USER_ARG - A value passed to SUCCESS_ROUTINE,
278 0346 1 1 ERROR_ROUTINE and CONFIRM_ROUTINE each
279 0347 1 1 time they are called. Whatever mechanism
280 0348 1 1 is used to pass USER_ARG to LIB$DELETE_FILE is
281 0349 1 1 used to pass it to the action routine. This
282 0350 1 1 is an optional parameter, if omitted, zero
283 0351 1 1 is passed by immediate value.
284 0352 1 1
285 0353 1 1 RESULTANT_NAME - A string into which is written the RMS resultant
286 0354 1 1 file specification of the last file processed
287 0355 1 1 by LIB$DELETE_FILE. Passed by descriptor.
288 0356 1 1 If present, RESULTANT_NAME is used to store
289 0357 1 1 the filespec passed to the action routines
290 0358 1 1 instead of a default class S, type T string.
291 0359 1 1
292 0360 1 1 FILE_SCAN_CONTEXT - The address of a longword, which is
293 0361 1 1 initialized to 0 before calling LIB$DELETE_FILE.
294 0362 1 1 This context is used by LIB$FILE_SCAN to retain
295 0363 1 1 multiple input file related file context, and
296 0364 1 1 need only be specified if dealing with multiple
297 0365 1 1 input files, as the DCL DELETE command does.
298 0366 1 1 The context allocated by LIB$FILE_SCAN while
299 0367 1 1 processing the LIB$DELETE_FILE requests may
300 0368 1 1 be deallocated by calling LIB$FILE_SCAN_END
301 0369 1 1 after all calls to LIB$DELETE_FILE have been
302 0370 1 1 completed.
303 0371 1 1
304 0372 1 1 IMPLICIT INPUTS:
305 0373 1 1
306 0374 1 1 NONE
307 0375 1 1
308 0376 1 1 IMPLICIT OUTPUTS:
309 0377 1 1
310 0378 1 1 NONE
311 0379 1 1
312 0380 1 1 COMPLETION STATUS:
313 0381 1 1
314 0382 1 1 SS$ NORMAL Normal successful completion
315 0383 1 1 LIB$_ERRROUCAL Success - error routine called. A file error was
316 0384 1 1 encountered but ERROR_ROUTINE was called to handle
317 0385 1 1 the condition.
318 0386 1 1 LIB$_INVSTRDES Invalid string descriptor
319 0387 1 1 LIB$_INVFILSPE Invalid file specification. FILESPEC or
320 0388 1 1 DEFAULT_FILESPEC is longer than 255 characters.
321 0389 1 1 LIB$_WRONUMARG Wrong number of arguments.
322 0390 1 1 LIB$_xxx Any error status from LIB$SCOPY xxx
323 0391 1 1 RM$_xxx Any error status from RMS. If ERROR_ROUTINE is not
324 0392 1 1 specified, this is the most severe of the RMS errors
325 0393 1 1 encountered while deleting the file(s).
326 0394 1 1
327 0395 1 1 SIDE EFFECTS:
328 0396 1 1
329 0397 1 1 The files specified are deleted.
330 0398 1 1
331 0399 1 1 --
332 0400 1 1
333 0401 2 BEGIN
```



```
334 0402 2
335 0403 2
336 0404 2
337 0405 2
338 0406 2
339 0407 2
340 0408 2
341 0409 2
342 0410 2
343 0411 2
344 0412 2
345 0413 2
346 0414 2
347 0415 2
348 0416 2
349 0417 2
350 0418 2
351 0419 2
352 0420 2
353 0421 2
354 0422 2
355 0423 2
356 0424 2
357 0425 2
358 0426 2
359 0427 2
360 0428 2
361 0429 2
362 0430 2
363 0431 2
364 0432 2
365 0433 2
366 0434 2
367 0435 2
368 0436 2
369 0437 2
370 0438 2
371 0439 2
372 0440 2
373 P 0441 2
374 0442 2
375 0443 2
376 0444 2
377 0445 2
378 0446 2
379 0447 2
380 0448 2
381 0449 2
382 0450 2
383 0451 2
384 0452 2
385 0453 2
386 0454 2
387 0455 2
388 0456 2
389 0457 2
390 0458 2

LOCAL
    FAB: $FAB_DECL,
    NAM: $NAM_DECL,
    RLF_NAM: $NAM_DECL,
    ESN: VECTOR [NAM$C_MAXRSS, BYTE],
    RSN: VECTOR [NAM$C_MAXRSS, BYTE],
    ERROR_SOURCE,

    WORST_ERROR,
    INTERCEPT_FLAG: VOLATILE,
    RET_STATUS;

BUILTIN
    ACTUALCOUNT,
    NULLPARAMETER;

!+
! Enable DELETE_HANDLER as a handler so that it will convert our
! signals to return statuses if INTERCEPT_FLAG is set.
!-

ENABLE
    DELETE_HANDLER (INTERCEPT_FLAG);

!+
! Validate the argument count.
!-

$LIB$VALIDATE_ARGCOUNT (1,9);

!+
! Initialize the FAB and NAM blocks.
!-

$FAB_INIT (FAB=FAB, NAM=NAM);
$NAM_INIT (NAM=NAM, ESA=ESN, ESS=NAM$C_MAXRSS,
    RSA=RSN, RSS=NAM$C_MAXRSS, RLF=RLF_NAM);
$NAM_INIT (NAM=RLF_NAM);

!+
! Initialize WORST_ERROR to zero. If any errors occur, the worst one
! will be in WORST_ERROR when LIB$FILE_SCAN returns.
! Initialize ERROR_SOURCE to zero. If DO_DELETE calls DELETE_ERROR
! it will set ERROR_SOURCE to 1.
!-

WORST_ERROR = 0;
ERROR_SOURCE = 0;

!+
! Set up FAB for the file specification. Return LIB$INVFLSPE
! if the string is longer than 255 characters.
!-
```

LIB\$DELETE\_FILE Delete one or more files  
1-006 LIB\$DELETE\_FILE - Delete one or more files

E 15  
16-Sep-1984 00:48:36  
14-Sep-1984 12:38:38

VAX-11 Bliss-32 V4.0-742  
[LIBRTL.SRC]LIBDELETE.B32;1

Page 8  
(3)

```

391      0459      2
392      0460      BEGIN
393      0461
394      0462      LOCAL
395      0463          FILESPEC_LENGTH: WORD,      ! Length of FILESPEC string
396      0464          STATUS;                      ! Status from LIB$ANALYZE_SDESC.
397      0465
398      0466      STATUS = LIB$ANALYZE_SDESC R2 (FILESPEC [0,0,0,0];
399      0467          FILESPEC_LENGTH, FAB [FAB$L_FNA]);
400      0468
401      0469      IF NOT .STATUS
402      0470      THEN
403      0471          RETURN .STATUS;
404      0472
405      0473      IF .FILESPEC_LENGTH GTRU 255
406      0474      THEN
407      0475          RETURN LIB$_INVFILSPE;
408      0476
409      0477      FAB [FAB$B_FNS] = .FILESPEC_LENGTH;
410      0478      END;
411      0479
412      0480      !+
413      0481      ! Get DEFAULT_FILESPEC if present, and set up as default name string.
414      0482      !-
415      0483
416      0484      IF NOT NULLPARAMETER (2)
417      0485      THEN
418      0486          BEGIN
419      0487
420      0488          LOCAL
421      0489              FILESPEC_LENGTH: WORD,      ! Length of DEFAULT FILESPEC string
422      0490              STATUS;                      ! Status from LIB$ANALYZE_SDESC.
423      0491
424      0492          STATUS = LIB$ANALYZE_SDESC R2 (DEFAULT_FILESPEC [0,0,0,0];
425      0493              FILESPEC_LENGTH, FAB [FAB$L_DNA]);
426      0494
427      0495          IF NOT .STATUS
428      0496          THEN
429      0497              RETURN .STATUS;
430      0498
431      0499          IF .FILESPEC_LENGTH GTRU 255
432      0500          THEN
433      0501              RETURN LIB$_INVFILSPE;
434      0502
435      0503          FAB [FAB$B_DNS] = .FILESPEC_LENGTH;
436      0504          END;
437      0505
438      0506      !+
439      0507      ! Get RELATED_FILESPEC if present, and set up as related name string.
440      0508      !-
441      0509
442      0510      IF NOT NULLPARAMETER (3)
443      0511          AND NULLPARAMETER (9)
444      0512      THEN
445      0513          BEGIN
446      0514
447      0515          LOCAL
```

LIB\$DELETE\_FILE Delete one or more files  
1-006 LIB\$DELETE\_FILE - Delete one or more files

F 15  
16-Sep-1984 00:48:36 VAX-11 Bliss-32 V4.0-742  
14-Sep-1984 12:38:38 [LIBRTL.SRC]LIBDELETE.B32;1

Page 9  
(3)

```

448      0516      FILESPEC_LENGTH: WORD,      ! Length of RELATED FILESPEC string
449      0517      STATUS;                      ! Status from LIB$ANALYZE_SDESC.
450      0518
451      0519      STATUS = LIB$ANALYZE_SDESC R2 (RELATED FILESPEC [0,0,0,0];
452      0520      FILESPEC_LENGTH, RLF_NAM [NAMS_L_RSL]);
453      0521
454      0522      IF NOT .STATUS
455      0523      THEN
456      0524      RETURN .STATUS;
457      0525
458      0526      IF .FILESPEC_LENGTH GTRU 255
459      0527      THEN
460      0528      RETURN LIB$_INVFILESPE;
461      0529
462      0530      RLF_NAM [NAMS_B_RSL] = .FILESPEC_LENGTH;
463      0531      END;
464      0532
465      0533
466      0534      !+ Call LIB$FILE_SCAN, which will call DO_DELETE for each file that is
467      0535      ! to be deleted. Pass as extra parameters the success routine, error routine,
468      0536      ! confirm routine, user argument, result filename descriptor, worst error
469      0537      ! holder, error source and intercept flag. These will be passed to DO_DELETE by
470      0538      ! LIB$FILE_SCAN.
471      0539      !-
472      0540
473      0541      LIB$FILE_SCAN (
474      0542      FAB,                      ! Input FAB
475      0543      DO_DELETE,              ! Success routine
476      0544      DELETE_ERROR,          ! Error routine
477      0545      (IF ACTUALCOUNT () GEQU 9 THEN .FILE_SCAN_CONTEXT ELSE 0), ! context
478      0546      (IF ACTUALCOUNT () GEQU 4 THEN .SUCCESS_ROUTINE ELSE 0), ! Success routine
479      0547      (IF ACTUALCOUNT () GEQU 5 THEN .ERROR_ROUTINE ELSE 0), ! Error routine
480      0548      (IF ACTUALCOUNT () GEQU 6 THEN .CONFIRM_ROUTINE ELSE 0), ! Confirm routine
481      0549      (IF ACTUALCOUNT () GEQU 7 THEN .USER_ARG ELSE 0), ! User argument
482      0550      (IF ACTUALCOUNT () GEQU 8 THEN .RESULTANT_NAME ELSE 0), ! Resultant name
483      0551      WORST_ERROR,          ! Worst error so far
484      0552      ERROR_SOURCE,         ! Code for source of error
485      0553      INTERCEPT_FLAG); ! Signal intercept flag
486      0554
487      0555
488      0556      !+ Parse the null string to deallocate any internal
489      0557      ! RMS context.
490      0558      !-
491      0559
492      0560      NAM[NAMSV_SVCTX] = 0;
493      0561      NAM[NAMSV_SYNCHK] = 1;
494      0562      NAM[NAMSB_ESL] = 0;
495      0563      NAM[NAMSB_RSL] = 0;
496      0564      NAM[NAMSB_ESS] = 0;
497      0565      NAM[NAMSB_RSS] = 0;
498      0566      NAM[NAMSL_RLF] = 0;
499      0567      FAB[FABSB_FNS] = 0;
500      0568      FAB[FABSB_DNS] = 0;
501      0569      $PARSE(FAB=FAB);
502      0570
503      0571
504      0572      !+ Return WORST_ERROR or $$$_NORMAL, as appropriate.
```



LIB\$DELETE\_FILE Delete one or more files  
1-006 LIB\$DELETE\_FILE - Delete one or more files

G 15  
16-Sep-1984 00:48:36  
14-Sep-1984 12:38:38

VAX-11 Bliss-32 V4.0-742  
[LIBRTL.SRC]LIBDELETE.B32;1

Page 10  
(3)

```

505      0573      1      !-
506      0574
507      0575      IF .WORST_ERROR NEQ 0
508      0576      THEN
509      0577          RET_STATUS = .WORST_ERROR
510      0578      ELSE
511      0579          RET_STATUS = $$$_NORMAL;
512      0580
513      0581      RETURN .RET_STATUS;
514      0582
515      0583      END;
```

! End of routine LIB\$DELETE\_FILE

.TITLE LIB\$DELETE\_FILE Delete one or more files  
.IDENT \1-006\

.EXTRN LIB\$ANALYZE\_SDESC\_R2  
.EXTRN LIB\$FILE\_SCAN, LIB\$SCOPY\_DXDX  
.EXTRN LIB\$SCOPY\_R\_DX, LIB\$SIG\_TO\_RET  
.EXTRN LIB\$\_ERRRUCAL, LIB\$\_INVFICSP  
.EXTRN LIB\$\_WRONUMARG, SY\$\_PARSE

.PSECT \_LIB\$CODE, NOWRT, SHR, PIC, 2

.ENTRY LIB\$DELETE\_FILE, Save R2,R3,R4,R5,R6  
MOVAB LIB\$ANALYZE\_SDESC\_R2, R6

MOVAB -796(SP), SP  
CLRL INTERCEPT\_FLAG  
MOVAL 22\$, (FP)  
SUBB3 #1, (AP), DIFF  
CMPB DIFF, #8  
BLEQU 1\$  
MOVL #LIB\$\_WRONUMARG, R0  
RET  
MOVC5 1\$:

MOVW #20483, \$RMS\_PTR  
MOVB #2, \$RMS\_PTR+22  
MOVB #2, \$RMS\_PTR+31  
MOVAB NAM, \$RMS\_PTR+40  
MOVC5 #0, (SP), #0, #96, \$RMS\_PTR

MOVW #24578, \$RMS\_PTR  
MNEGB #1, \$RMS\_PTR+2  
MOVAB RSN, \$RMS\_PTR+4  
MNEGB #1, \$RMS\_PTR+10  
MOVAB ESN, \$RMS\_PTR+12  
MOVAB RLF\_NAM, \$RMS\_PTR+16  
MOVC5 #0, (SP), #0, #96, \$RMS\_PTR

MOVW #24578, \$RMS\_PTR  
CLRG ERROR\_SOURCE  
MOVL FILESPEC, R0  
JSB LIB\$ANALYZE\_SDESC\_R2  
MOVL R2, FAB+44  
BLBC STATUS, 4\$  
CMPW FILESPEC\_LENGTH, #255

```

56 00000000G 00 007C 00000
5E FCE4 CE 9E 00002
      08 AE D4 0000E
6D 0176 CF DE 00011
50 6C 01 83 00016
      08 50 91 0001A
      08 1B 0001D
50 00000000G 8F D0 0001F
      04 00026
0050 8F 00 6E 00 2C 00027 1$:
      B0 AD 5003 8F B0 00030
      C6 AD 02 90 00036
      CF AD 02 90 0003A
      DB AD FF50 CD 9E 0003E
0060 8F 00 6E 00 2C 00044
      FF50 CD 0004B
      FF52 CD 6002 8F B0 0004E
      FF54 CD 0C AE 9E 0005A
      FF5A CD 01 8E 00060
      FF5C CD 010C CE 9E 00065
0060 8F 00 6E 00 2C 00073
      FEFO CD 0007A
      FEFO CD 6002 8F B0 0007D
      50 04 6E 7C 00084
      DC AD 66 16 0008A
      4C 52 D0 0008C
00FF 8F 51 B1 00093
```

```

: 0175
:
: 0401
: 0434
:
: 0440
:
: 0442
:
: 0443
: 0453
: 0466
: 0467
: 0469
: 0473
```

E4	AD	50	1A	00098	BGTRU	6\$		
	02	51	90	0009A	MOVB	FILESPEC_LENGTH, FAB+52		0477
		6C	91	0009E	CMPB	(AP), #2		0484
		1D	1F	000A1	BLSSU	2\$		
		08	AC	D5	TSTL	8(AP)		
		18	13	000A6	BEQL	2\$		
	50	08	AC	D0	MOVL	DEFAULT_FILESPEC, R0		0492
		66	16	000AC	JSB	LIB\$ANALYZE_SDESC_R2		
E0	AD	52	D0	000AE	MOVL	R2, FAB+48		0493
00FF	2A	50	E9	000B2	BLBC	STATUS, 4\$		0495
	8F	51	B1	000B5	CMPW	FILESPEC_LENGTH, #255		0499
		2E	1A	000BA	BGTRU	6\$		
E5	AD	51	90	000BC	MOVB	FILESPEC_LENGTH, FAB+53		0503
	03	6C	91	000C0	CMPB	(AP), #3		0510
		32	1F	000C3	BLSSU	8\$		
		0C	AC	D5	TSTL	12(AP)		
		2D	13	000C8	BEQL	8\$		
	09	6C	91	000CA	CMPB	(AP), #9		0511
		05	1F	000CD	BLSSU	3\$		
		24	AC	D5	TSTL	36(AP)		
		23	12	000D2	BNEQ	8\$		
	50	0C	AC	D0	MOVL	RELATED_FILESPEC, R0		0519
		66	16	000D8	JSB	LIB\$ANALYZE_SDESC_R2		
FEF4	CD	52	D0	000DA	MOVL	R2, RLF_NAM+4		0520
	01	50	E8	000DF	BLBS	STATUS, 5\$		0522
			04	000E2	RET			
00FF	8F	51	B1	000E3	CMPW	FILESPEC_LENGTH, #255		0526
		08	1B	000E8	BLEQU	7\$		
	50	8F	D0	000EA	MOVL	#LIB\$_INVFILSPE, R0		0528
			04	000F1	RET			
FEF3	CD	51	90	000F2	MOVB	FILESPEC_LENGTH, RLF_NAM+3		0530
		08	AE	9F	PUSHAB	INTERCEPT_FLAG		0541
		04	AE	9F	PUSHAB	ERROR_SOURCE		
		0C	AE	9F	PUSHAB	WORST_ERROR		
	08	6C	91	00100	CMPB	(AP), #8		0550
		05	1F	00103	BLSSU	9\$		
		20	AC	DD	PUSHL	RESULTANT_NAME		
		02	11	00108	BRB	10\$		
		7E	D4	0010A	CLRL	-(SP)		
	07	6C	91	0010C	CMPB	(AP), #7		0549
		05	1F	0010F	BLSSU	11\$		
		1C	AC	DD	PUSHL	USER_ARG		
		02	11	00114	BRB	12\$		
		7E	D4	00116	CLRL	-(SP)		
	06	6C	91	00118	CMPB	(AP), #6		0548
		05	1F	0011B	BLSSU	13\$		
		18	AC	DD	PUSHL	CONFIRM_ROUTINE		
		02	11	00120	BRB	14\$		
		7E	D4	00122	CLRL	-(SP)		
	05	6C	91	00124	CMPB	(AP), #5		0547
		05	1F	00127	BLSSU	15\$		
		14	AC	DD	PUSHL	ERROR_ROUTINE		
		02	11	0012C	BRB	16\$		
		7E	D4	0012E	CLRL	-(SP)		
	04	6C	91	00130	CMPB	(AP), #4		0546
		05	1F	00133	BLSSU	17\$		
		10	AC	DD	PUSHL	SUCCESS_ROUTINE		

LIB\$DELETE\_FILE Delete one or more files  
1-006 LIB\$DELETE\_FILE - Delete one or more files

I 15  
16-Sep-1984 00:48:36  
14-Sep-1984 12:38:38

VAX-11 Bliss-32 V4.0-742  
[LIBRTL.SRC]LIBDELETE.B32;1

Page 12  
(3)

		02	11	00138	BRB	18\$	
		7E	D4	0013A	CLRL	-(SP)	
	09	6C	91	0013C	CMPB	(AP), #9	0545
		05	1F	0013F	BLSSU	19\$	
	24	AC	DD	00141	PUSHL	FILE_SCAN_CONTEXT	
		02	11	00144	BRB	20\$	
		7E	D4	00146	CLRL	-(SP)	
	0000V	CF	9F	00148	PUSHAB	DELETE_ERROR	0541
	0000V	CF	9F	0014C	PUSHAB	DO_DELETE	
	B0	AD	9F	00150	PUSHAB	FAB	
00000000G	00	0C	FB	00153	CALLS	#12, LIB\$FILE_SCAN	0560
83	AD	8F	8A	0015A	BICB2	#128, NAM+51	0561
FF58	CD	08	88	0015F	BISB2	#8, NAM+8	0565
		CD	B4	00164	CLRW	NAM+2	0564
	FF52	CD	B4	00168	CLRW	NAM+10	0566
	FF5A	CD	D4	0016C	CLRL	NAM+16	0567
	FF60	CD	D4	00170	CLRW	FAB+52	0569
	E4	AD	B4	00173	PUSHAB	FAB	
	B0	AD	9F	00176	CALLS	#1, SYSSPARSE	
00000000G	00	01	FB	0017D	TSTL	WORST_ERROR	0575
		04	AE	D5	BEQL	21\$	
	50	04	AE	D0	MOVL	WORST_ERROR, RET_STATUS	0577
				04	RET		
	50		01	D0	MOVL	#1, RET_STATUS	0579
				04	RET		0583
				0000	.WORD	Save nothing	0401
	50	08	AC	D0	MOVL	8(AP), R0	
	50	04	A0	D0	MOVL	4(R0), R0	
		FCEC	C0	9F	PUSHAB	INTERCEPT_FLAG	
			01	DD	PUSHL	#1	
			5E	DD	PUSHL	SP	
	7E	04	AC	7D	MOVQ	4(AP), -(SP)	
0000V	CF	03	FB	001A1	CALLS	#3, DELETE_HANDLER	
			04	001A6	RET		

; Routine Size: 423 bytes, Routine Base: \_LIB\$CODE + 0000



```
517 0584 1 %SBTTL 'DO_DELETE - Delete a file'
518 0585 1 ROUTINE DO_DELETE (
519 0586 1   FAB: REF $FAB_DECL,
520 0587 1   UNUSED_1,
521 0588 1   UNUSED_2,
522 0589 1   UNUSED_3,
523 0590 1   SUCCESS_ROUTINE,
524 0591 1   UNUSED_4,
525 0592 1   CONFIRM_ROUTINE,
526 0593 1   USER_ARG,
527 0594 1   RESULTANT_NAME,
528 0595 1   UNUSED_5,
529 0596 1   ERROR_SOURCE: REF VECTOR [, LONG],
530 0597 1   INTERCEPT_FLAG: REF VECTOR [, LONG]
531 0598 1 ): NOVALUE =
532 0599 1
533 0600 1 ++
534 0601 1 FUNCTIONAL DESCRIPTION:
535 0602 1
536 0603 1   This routine is called once for each file that LIB$FILE_SCAN
537 0604 1   finds. It deletes the file described by FAB.
538 0605 1
539 0606 1 CALLING SEQUENCE:
540 0607 1
541 0608 1   DO_DELETE is called with the same arguments as was LIB$FILE_SCAN
542 0609 1   in LIB$DELETE_FILE.
543 0610 1
544 0611 1 FORMAL PARAMETERS:
545 0612 1
546 0613 1   See LIB$DELETE_FILE for a description of all parameters.
547 0614 1
548 0615 1 IMPLICIT INPUTS:
549 0616 1
550 0617 1   NONE
551 0618 1
552 0619 1 IMPLICIT OUTPUTS:
553 0620 1
554 0621 1   NONE
555 0622 1
556 0623 1 COMPLETION STATUS:
557 0624 1
558 0625 1   NONE
559 0626 1
560 0627 1 SIDE EFFECTS:
561 0628 1
562 0629 1   Deletes a file.
563 0630 1
564 0631 1 --
565 0632 1
566 0633 2 BEGIN
567 0634 2
568 0635 2 LOCAL
569 0636 2   NAM: REF $NAM_DECL,
570 0637 2   LOCAL_DSC: BLOCK [8, BYTE],
571 0638 2   STRING_PTR,
572 0639 2   COPY_STATUS;
573 0640 2
```

```
! Next input file FAB
! Unused here
! Unused here
! Unused here
! Success routine address
! Unused here
! Confirm routine address
! User argument
! Resultant filename
! Unused here
! Source of error
! Intercept flag
```

```
! NAM of FAB
! Local string descriptor
! Pointer to string used
! Status from LIB$COPY
```

```
574 0641 2 BUILTIN
575 0642 2 AP,
576 0643 2 CALLG;
577 0644 2
578 0645 2 !+
579 0646 2 | Set NAM address.
580 0647 2 |
581 0648 2 |
582 0649 2 NAM = .FAB [FAB$L_NAM];
583 0650 2
584 0651 2 !+
585 0652 2 | Copy the resultant file name to the user's string, if specified.
586 0653 2 | If we get an error from the copy, signal it.
587 0654 2 | If no user string, use our own.
588 0655 2 |
589 0656 2 |
590 0657 2 STRING_PTR = .RESULTANT_NAME;
591 0658 2 IF .STRING_PTR NEQA 0
592 0659 2 THEN
593 0660 2 BEGIN
594 0661 2 COPY_STATUS = LIB$SCOPY R DX (%REF(.NAM [NAM$B_RSL]),
595 0662 2 .NAM [NAM$L_RSA], .RESULTANT_NAME);
596 0663 2 IF NOT .COPY_STATUS
597 0664 2 THEN
598 0665 2 BEGIN
599 0666 2 INTERCEPT_FLAG [0] = 1;
600 0667 2 SIGNAL_STOP (.COPY_STATUS);
601 0668 2 RETURN;
602 0669 2 END;
603 0670 2 END
604 0671 2 ELSE
605 0672 2 BEGIN
606 0673 2 STRING_PTR = LOCAL_DSC; ! Use our string
607 0674 2 LOCAL_DSC [DSC$B_DTYPE] = DSC$K_DTYPE_T;
608 0675 2 LOCAL_DSC [DSC$B_CLASS] = DSC$K_CLASS_S;
609 0676 2 LOCAL_DSC [DSC$W_LENGTH] = .NAM [NAM$B_RSL];
610 0677 2 LOCAL_DSC [DSC$A_POINTER] = .NAM [NAM$L_RSA];
611 0678 2 END;
612 0679 2
613 0680 2 !+
614 0681 2 | If there is a CONFIRM_ROUTINE, ask it if it wants the file deleted.
615 0682 2 | Note use of linkage name BLISS to perform a general-routine-call.
616 0683 2 |
617 0684 2 |
618 0685 2 IF .CONFIRM_ROUTINE NEQA 0
619 0686 2 THEN
620 0687 2 IF NOT BLISS (.CONFIRM_ROUTINE, .STRING_PTR, .FAB, .USER_ARG)
621 0688 2 THEN
622 0689 2 RETURN;
623 0690 2
624 0691 2 !+
625 0692 2 | Do the $ERASE.
626 0693 2 |
627 0694 2 |
628 0695 2 IF NOT $ERASE (FAB = FAB [0,0,0,0])
629 0696 2 THEN
630 0697 2 BEGIN
```

```
631 0698      +
632 0699      | Call the DELETE_ERROR routine with the same arguments we were
633 0700      | called with. It will call the user's error routine, if any.
634 0701      | Set ERROR_SOURCE to 1 so that DELETE_ERROR will know the error
635 0702      | came from the $ERASE.
636 0703      |
637 0704      | ERROR_SOURCE [0] = 1;
638 0705      | CALLG (.AP, DELETE_ERROR);
639 0706      | RETURN;
640 0707      | END;
641 0708
642 0709      +
643 0710      | If there is a success routine, call it.
644 0711      |
645 0712      |
646 0713      | IF .SUCCESS_ROUTINE NEQA 0
647 0714      | THEN
648 0715      |     BLISS (.SUCCESS_ROUTINE, .STRING_PTR, .USER_ARG);
649 0716
650 0717      | RETURN;
651 0718
652 0719      | END;
                                ! End of routine DO_DELETE
```

.EXTRN SYS\$ERASE

```
001C 00000 DO_DELETE:
5E      0C C2 00002      .WORD      Save R2,R3,R4      0585
53      04 AC D0 00005      SUBL2      #12, SP
52      28 A3 D0 00009      MOVL       FAB, R3      0649
54      24 AC D0 0000D      MOVL       40(R3), NAM
26      13 00011      MOVL       RESULTANT_NAME, STRING_PTR      0657
24      AC DD 00013      BEQL        1$      0658
04      A2 DD 00016      PUSHL      RESULTANT_NAME      0662
03      A2 9A 00019      PUSHL      4(NAM)
08      AE 9F 0001E      MOVZBL     3(NAM), 8(SP)      0661
00000000G 00      03 FB 00021      PUSHAB  8(SP)
22      50 E8 00028      CALLS      #3, LIB$SCOPY_R_DX
30      BC      01 D0 0002B      BLBS      COPY_STATUS, 2$      0663
00000000G 00      50 DD 0002F      MOVL       #1, $INTERCEPT_FLAG      0666
01      FB 00031      PUSHL      COPY_STATUS      0667
04      04 00038      CALLS      #1, LIB$STOP
54      04 AE 9E 00039 1$:      RET      0665
06      AE 010E 8F B0 0003D      MOVAB      LOCAL_DSC, STRING_PTR      0673
04      AE 03 A2 9B 00043      MOVW      #270, LOCAL_DSC+2      0674
08      AE 04 A2 D0 00048      MOVZBW   3(NAM), LOCAL_DSC      0676
1C      AC D5 0004D 2$:      MOVL       4(NAM), LOCAL_DSC+4      0677
0E      13 00050      TSTL        CONFIRM_ROUTINE      0685
20      AC DD 00052      BEQL        3$
53      DD 00055      PUSHL      USER_ARG      0687
54      DD 00057      PUSHL      R3
1C      BC      03 FB 00059      PUSHL      STRING_PTR
24      50 E9 0005D      CALLS      #3, $CONFIRM_ROUTINE
00000000G 00      53 DD 00060 3$:      BLBC      R0, 5$
01      FB 00062      PUSHL      R3      0695
CALLS      #1, SYS$ERASE
```



LIB\$DELETE\_FILE Delete one or more files  
1-006 DO\_DELETE - Delete a file

M 15  
16-Sep-1984 00:48:36  
14-Sep-1984 12:38:38

VAX-11 Bliss-32 V4.0-742  
[LIBRTL.SRC]LIBDELETE.B32;1

Page 16  
(4)

2C 0A  
0000V BC  
CF

50 E8 00069  
01 D0 0006C  
6C FA 00070  
04 00075  
14 AC D5 00076 4\$:  
09 13 00079  
20 AC DD 0007B  
54 DD 0007E  
02 FB 00080  
04 00084 5\$:

BLBS R0, 4\$  
MOVL #1, @ERROR\_SOURCE  
CALLG (AP), DELETE\_ERROR  
RET  
TSTL SUCCESS\_ROUTINE  
BEQL 5\$  
PUSHL USER\_ARG  
PUSHL STRING\_PTR  
CALLS #2, @SUCCESS\_ROUTINE  
RET

0704  
0705  
0697  
0713  
0715  
0719

; Routine Size: 133 bytes, Routine Base: \_LIB\$CODE + 01A7

```

654 0720 1 XSBTTL 'DELETE_ERROR - Report error during delete'
655 0721 1 ROUTINE DELETE_ERROR (
656 0722 1     FAB: REF $FAB_DECL,           ! Next input file FAB
657 0723 1     UNUSED_1,                 ! Unused here
658 0724 1     UNUSED_2,                 ! Unused here
659 0725 1     UNUSED_3,                 ! Unused here
660 0726 1     UNUSED_4,                 ! Unused here
661 0727 1     ERROR_ROUTINE,           ! User's error routine
662 0728 1     UNUSED_5,                 ! Unused here
663 0729 1     USER_ARG,                ! User's argument
664 0730 1     RESULTANT_NAME,          ! Resultant file name
665 0731 1     WORST_ERROR: REF BLOCK [, BYTE], ! Worst previous error
666 0732 1     ERROR_SOURCE: REF VECTOR [, LONG], ! Code for source of error
667 0733 1     INTERCEPT_FLAG: REF VECTOR [, LONG] ! Signal intercept flag
668 0734 1 ): NOVALUE =
669 0735 1
670 0736 1 ++
671 0737 1 FUNCTIONAL DESCRIPTION:
672 0738 1
673 0739 1     This routine is called when LIB$FILE_SCAN detects an error.
674 0740 1     It calls the user's error routine, if one exists.
675 0741 1     If the user's error routine returns success, or if there is
676 0742 1     no user error routine, processing of remaining files continues.
677 0743 1     If the user's error routine returns failure, the error is
678 0744 1     signalled. This signal is converted to the return status of
679 0745 1     LIB$DELETE_FILE.
680 0746 1
681 0747 1 CALLING SEQUENCE:
682 0748 1
683 0749 1     Called from LIB$FILE_SCAN and DO DELETE with the same arguments
684 0750 1     as passed to LIB$FILE_SCAN from LIB$DELETE_FILE.
685 0751 1
686 0752 1 FORMAL PARAMETERS:
687 0753 1
688 0754 1     See body of LIB$DELETE_FILE for descriptions of arguments.
689 0755 1
690 0756 1 IMPLICIT INPUTS:
691 0757 1
692 0758 1     NONE
693 0759 1
694 0760 1 IMPLICIT OUTPUTS:
695 0761 1
696 0762 1     NONE
697 0763 1
698 0764 1 COMPLETION STATUS:
699 0765 1
700 0766 1     NONE
701 0767 1
702 0768 1 SIDE EFFECTS:
703 0769 1
704 0770 1     Signals .FAB [FAB$L_STS]
705 0771 1
706 0772 1 --
707 0773 1
708 0774 2 BEGIN
709 0775 2
710 0776 2 LOCAL
  
```

LIB\$DELETE\_FILE Delete one or more files  
1-006 DELETE\_ERROR - Report error during delete

8 16  
16-Sep-1984 00:48:36 VAX-11 Bliss-32 V4.0-742  
14-Sep-1984 12:38:38 [LIBRTL.SRC]LIBDELETE.B32;1

Page 18  
(5)

```

711      0777      2      NAM: REF $NAM DECL      ! NAM block
712      0778      LOCAL_DSC: BLOCK [8, BYTE],      ! Our filename descriptor
713      0779      STRING_PTR,      ! Pointer to user's string or our own
714      0780      COPY_STATUS;      ! Status from LIB$SCOPY
715      0781
716      0782
717      0783      !+ Build our local descriptor to contain the filespec, the file in error.
718      0784      ! Note that the filespec is guaranteed to be in the h.w. because
719      0785      ! LIB$FILE_SCAN puts it there before calling us.
720      0786
721      0787
722      0788      NAM = .FAB [FAB$L_NAM];
723      0789      LOCAL_DSC [DSC$B_DTYPE] = DSC$K_DTYPE_T;
724      0790      LOCAL_DSC [DSC$B_CLASS] = DSC$K_CLASS_S;
725      0791      LOCAL_DSC [DSC$W_LENGTH] = .NAM [NAM$B_RSL];
726      0792      LOCAL_DSC [DSC$A_POINTER] = .NAM [NAM$C_RSA];
727      0793
728      0794
729      0795      !+ If the user has specified DELETED_FILE, copy the filename to it.
730      0796
731      0797
732      0798      STRING_PTR = .RESULTANT_NAME;
733      0799      IF .STRING_PTR NEQA 0
734      0800      THEN
735      0801          BEGIN
736      0802              COPY_STATUS = LIB$SCOPY_DXDX (LOCAL_DSC, .RESULTANT_NAME);
737      0803              IF NOT .COPY_STATUS
738      0804              THEN
739      0805                  BEGIN
740      0806                      INTERCEPT_FLAG [0] = 1;      ! Cause handler to intercept signal
741      0807                      SIGNAL_STOP (.COPY_STATUS);
742      0808                      RETURN;
743      0809                      END;
744      0810                  END
745      0811      ELSE
746      0812          STRING_PTR = LOCAL_DSC;      ! Use our own string
747      0813
748      0814
749      0815      !+ If a user error routine has been specified, call it with arguments
750      0816      ! of the filename, STS, STV, error source and user argument. If it returns
751      0817      ! failure, signal the error. Since the user's routine was called,
752      0818      ! store LIB$ERRROUCAL in WORST_ERROR.
753      0819
754      0820
755      0821      IF .ERROR_ROUTINE NEQA 0
756      0822      THEN
757      0823          BEGIN
758      0824              IF NOT BLISS (.ERROR_ROUTINE, .STRING_PTR,
759      0825                  FAB [FAB$L_STS], FAB [FAB$L_STV], .ERROR_SOURCE [0], .USER_ARG)
760      0826              THEN
761      0827                  BEGIN
762      0828                      INTERCEPT_FLAG [0] = 1;      ! Cause handler to intercept signal
763      0829                      SIGNAL_STOP (.FAB [FAB$L_STS]);
764      0830                      RETURN;
765      0831                      END;
766      0832                  WORST_ERROR [0,0,32,0] = LIB$ERRROUCAL;      ! Error routine called
767      0833                  END
```



```
768 0834 ELSE
769 0835 BEGIN
770 0836
771 0837     If this error is worse than any previous errors, store it in
772 0838     WORST_ERROR. (Use GEQU so that the initial zero gets replaced if
773 0839     the error is a warning.)
774 0840
775 0841
776 0842     IF .BLOCK [FAB [FAB$L_STS], ST$V_SEVERITY;4, BYTE] GEQU
777 0843     .WORST_ERROR [ST$V_SEVERITY]
778 0844 THEN
779 0845     WORST_ERROR [0,0,32,0] = .FAB [FAB$L_STS];
780 0846 END;
781 0847
782 0848
783 0849     + Reset ERROR_SOURCE to zero so that future $SEARCH errors will be
784 0850     properly indicated.
785 0851
786 0852
787 0853 ERROR_SOURCE [0] = 0;
788 0854
789 0855 RETURN;
790 0856
791 0857 END;
```

! End of routine DELETE\_ERROR

```
000C 00000 DELETE_ERROR:
08 C2 00002 .WORD Save R2,R3
04 AC D0 00005 SUBL2 #8, SP
28 A2 D0 00009 MOVL FAB, R2
02 AE 010E 8F B0 0000D MOVL 40(R2), NAM
04 AE 03 A0 9B 00013 MOVW #270, LOCAL_DSC+2
53 04 A0 D0 00017 MOVZBW 3(NAM), LOCAL_DSC
24 AC D0 0001C MOVL 4(NAM), LOCAL_DSC+4
18 13 00020 MOVL RESULTANT_NAME, STRING_PTR
24 AC DD 00022 BEQL 1$
04 AE 9F 00025 PUSHL RESULTANT_NAME
02 FB 00028 PUSHAB LOCAL_DSC
30 BC 50 E8 0002F CALLS #2, LIB$COPY_DXDX
53 01 D0 00032 BLBS COPY_STATUS, 2$
24 11 00036 MOVL #1, $INTERCEPT_FLAG
18 6E 9E 0003A BRB 3$
20 AC D5 0003D MOVAB LOCAL_DSC, STRING_PTR
2C AC DD 00042 TSTL ERROR_ROUTINE
0C A2 9F 00048 BEQL 5$
08 A2 9F 0004B PUSHL USER_ARG
53 DD 0004E PUSHL ERROR_SOURCE
18 BC 05 FB 00050 PUSHAB 12(R2)
30 BC 50 E8 00054 PUSHAB 8(R2)
01 D0 00057 PUSHL STRING_PTR
CALLS #5, $ERROR_ROUTINE
BLBS R0, 4$
MOVL #1, $INTERCEPT_FLAG
```

0721

0788

0789

0791

0792

0798

0799

0802

0803

0806

0807

0812

0821

0825

0828

LIB\$DELETE\_FILE Delete one or more files  
 1-006 DELETE\_ERROR - Report error during delete

D 16  
 16-Sep-1984 00:48:36 YAX-11 Bliss-32 V4.0-742  
 14-Sep-1984 12:38:38 [LIBRTL.SRC]LIBDELETE.B32;1

Page 20  
 (5)

				08	A2	DD	00058		PUSHL	8(R2)		0829
		00000000G	00		01	FB	0005E	38:	CALLS	#1, LIB\$STOP		0827
						04	00065		RET			0832
		28	BC	00000000G	8F	D0	00066	48:	MOVL	#LIB\$ERRROUCAL, @WORST_ERROR		0821
					13	11	0006E		BRB	68		0843
50	28	BC		03	00	EF	00070	58:	EXTZV	#0, #3, @WORST_ERROR, R0		
50	08	A2		03	00	ED	00076		CMPZV	#0, #3, 8(R2), -R0		
					05	1F	0007C		BLSSU	68		
		28	BC		08	A2	D0	0007E	MOVL	8(R2), @WORST_ERROR		0845
				2C	BC	D4	00083	68:	CLRL	@ERROR_SOURCE		0853
						04	00086		RET			0857

; Routine Size: 135 bytes, Routine Base: \_LIB\$CODE + 022C

LIB\$DELETE\_FILE Delete one or more files  
1-006 DELETE\_HANDLER - Local condition handler

E 16  
16-Sep-1984 00:48:36  
14-Sep-1984 12:38:38

VAX-11 Bliss-32 V4.0-742  
[LIBRTL.SRC]LIBDELETE.B32;1

Page 21  
(6)

```

793 0858 1 %SBTTL 'DELETE_HANDLER - Local condition handler'
794 0859 1 ROUTINE DELETE_HANDLER (
795 0860 1     SIGNAL_ARGS: REF BLOCK [, BYTE],      ! Signal arguments array
796 0861 1     MECH_ARGS: REF BLOCK [, BYTE],      ! Mechanism arguments array
797 0862 1     ENABLE_ARGS: REF VECTOR [, LONG]   ! Enable arguments array
798 0863 1 ) =
799 0864 1
800 0865 1 ++
801 0866 1 FUNCTIONAL DESCRIPTION:
802 0867 1
803 0868 1     This is the condition handler enabled by LIB$DELETE_FILE.
804 0869 1     If this is not an unwind, and if the INTERCEPT_FLAG enable
805 0870 1     argument is set, then LIB$SIG_TO_RET is called to convert the
806 0871 1     signal to a return status.
807 0872 1
808 0873 1 CALLING SEQUENCE:
809 0874 1
810 0875 1     status.wlc.v = DELETE_HANDLER (SIGNAL_ARGS.rl.ra, MECH_ARGS.rl.ra
811 0876 1     , ENABLE_ARGS.rl.ra)
812 0877 1
813 0878 1 FORMAL PARAMETERS:
814 0879 1
815 0880 1     SIGNAL_ARGS    - The signal argument list.
816 0881 1
817 0882 1     MECH_ARGS      - The mechanism argument list.
818 0883 1
819 0884 1     ENABLE_ARGS    - The enable argument list. The one enable
820 0885 1     argument is the address of INTERCEPT_FLAG;
821 0886 1
822 0887 1 IMPLICIT INPUTS:
823 0888 1
824 0889 1     NONE
825 0890 1
826 0891 1 IMPLICIT OUTPUTS:
827 0892 1
828 0893 1     NONE
829 0894 1
830 0895 1 ROUTINE VALUE:
831 0896 1
832 0897 1     SS$_RESIGNAL
833 0898 1
834 0899 1 SIDE EFFECTS:
835 0900 1
836 0901 1     May cause an unwind.
837 0902 1
838 0903 1 --
839 0904 1
840 0905 1 BEGIN
841 0906 1
842 0907 1 BUILTIN
843 0908 1     AP,      ! Argument pointer
844 0909 1     CALLG;  ! CALLG instruction
845 0910 1
846 0911 1 ++
847 0912 1     Determine if this is an unwind. If not, then if INTERCEPT_FLAG
848 0913 1     is set, turn this signal into an unwind.
849 0914 1 --
```



LIB\$DELETE\_FILE Delete one or more files  
1-006 DELETE\_HANDLER - Local condition handler

F 16  
16-Sep-1984 00:48:36  
14-Sep-1984 12:38:38

VAX-11 Bliss-32 V4.0-742  
[LIBRTL.SRC]LIBDELETE.B32;1

Page 22  
(6)

```

: 850      0915 2
: 851      0916 2      IF .SIGNAL_ARGS [CHF$L_SIG_NAME] NEQU SSS_UNWIND
: 852      0917 2      THEN
: 853      0918 2          IF ..ENABLE_ARGS [1]      ! Is INTERCEPT_FLAG set?
: 854      0919 2          THEN
: 855      0920 2              CALLG (.AP, LIB$SIG_TO_RET);      ! Convert signal to return status
: 856      0921 2          RETURN SSS_RESIGNAL;      ! Resignal error
: 857      0922 2
: 858      0923 2      END;      ! End of routine DELETE_HANDLER
: 859      0924 1
```

```

                                0000 00000 DELETE_HANDLER:
                                .WORD      Save nothing
00000920      50      04      AC      D0 00002      MOVL      SIGNAL_ARGS, R0
                                8F      04      AO      D1 00006      CMPL      4(R0), #2336
                                OF      13 0000E      BEQL      1$
                                50      0C      AC      D0 00010      MOVL      ENABLE_ARGS, R0
                                07      04      BO      E9 00014      BLBC      @4(R0), 1$
00000000G      00      6C      FA 00018      CALLG      (AP), LIB$SIG_TO_RET
                                50      0918 8F      3C 0001F 1$:      MOVZWL      #2328, R0
                                04 00024      RET
```

```

: 0859
: 0916
:
: 0918
:
: 0920
: 0922
: 0924
```

; Routine Size: 37 bytes, Routine Base: \_LIB\$CODE + 02B3

; 860 0925 1

LIB\$DELETE\_FILE Delete one or more files  
1-006 DELETE\_HANDLER - Local condition handler

G 16  
16-Sep-1984 00:48:36 VAX-11 Bliss-32 V4.0-742  
14-Sep-1984 12:38:38 [LIBRTL.SRC]LIBDELETE.B32;1

Page 23  
(7)

: 862 0926 1 END  
: 863 0927 1  
: 864 0928 0 ELUDOM

! End of module LIB\$DELETE\_FILE

.EXTRN LIB\$STOP

PSECT SUMMARY

: Name Bytes Attributes  
: \_LIB\$CODE 728 NOVEC,NOWRT, RD , EXE, SHR, LCL, REL, CON, PIC,ALIGN(2)

Library Statistics

File	----- Total	Symbols Loaded	----- Percent	Pages Mapped	Processing Time
\$255\$DUA28:[SYSLIB]STARLET.L32;1	9776	74	0	581	00:00.7
-\$255\$DUA28:[LIBRTL.OBJ]RTL.LIB.L32;1	36	2	5	8	00:00.1

COMMAND QUALIFIERS

: BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/NOTRACE/LIS=LISS:LIBDELETE/OBJ=OBJ\$:LIBDELETE MSRC\$:LIBDELETE/UPDATE=(ENHS:LIBDELETE  
: )

: Size: 728 code + 0 data bytes  
: Run Time: 00:12.4  
: Elapsed Time: 00:52.2  
: Lines/CPU Min: 4486  
: Lexemes/CPU-Min: 33065  
: Memory Used: 172 pages  
: Compilation Complete



0205 AH-BT13A-SE  
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION  
CONFIDENTIAL AND PROPRIETARY

LIBCUTPU  
LIS

LIBCUTUO  
LIS

LIBDAY  
LIS

LIBDECOVE  
LIS

LIBCUTOU  
LIS

LIBCUTPTU  
LIS

LIBCUTTPU  
LIS

LIBDECOFF  
LIS

LIBDOCOM  
LIS

LIBCUTTPZ  
LIS

LIBDATETI  
LIS

LIBDELETE  
LIS

LIBCUTPTO  
LIS

LIBDAYWK  
LIS

LIBCUTPTZ  
LIS